



Welcome United States Patent and Trademark Office

☐ Search Session History
[BROWSE](#)[SEARCH](#)[IEEE XPLORE GUIDE](#)

Fri, 1 Sep 2006, 5:13:19 PM EST

Edit an existing query or compose a new query in the Search Query Display.

Search Query Display

Select a search number (#) to:

- Add a query to the Search Query Display
- Combine search queries using AND, OR, or NOT
- Delete a search
- Run a search

Recent Search Queries

- | | |
|---------------------------|--|
| <u>#1</u> | ((multiprocessor or (multiprocessing unit))<in>metadata) |
| <u>#2</u> | cluster or node* |
| <u>#3</u> | (coherency and directory and consisten*<IN>metadata) |
| <u>#4</u> | (one-to-one <IN>metadata) |
| <u>#5</u> | (cross* interrogat*<IN>metadata) |
| <u>#6</u> | ((multiprocessor or (multiprocessing unit))<in>metadata) <AND> (cluster or node*) |
| <u>#7</u> | ((coherency and directory and consisten*<IN>metadata) <AND> ((one-to-one <IN>metadata)) |
| <u>#8</u> | ((multiprocessor or (multiprocessing unit))<in>metadata) <AND> (cluster or node*) <AND> ((one-to-one <IN>metadata)) |
| <u>#9</u> | ((coherency and directory and consisten*<IN>metadata) <AND> (((multiprocessor or (multiprocessing unit))<in>metadata) <AND> (cluster or node*)) <AND> ((one-to-one <IN>metadata))) |

Indexed by
 Inspec[®]

[Help](#) [Contact Us](#) [Privacy](#)

© Copyright 2006 IE

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	3206	(cluster\$2 or node\$2) same multiprocess\$6	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:32
S2	79894	"one to one" or "one-to-one" or "1 to one" or "1-to-one" or "one to 1" or "one-to-1" or "point to point" or "point-to-point" or "1 to 1" or "1-to-1"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:34
S3	129346	coheren\$6	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:35
S4	710	S1 and S2	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:35
S5	404	S3 and S4	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:35
S6	2	"20040268052".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 08:00
S7	24270902	@ad<"20030627"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:41
S8	328	S5 and S7	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:41
S9	0	711/*..ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:42

EAST Search History

S10	29385	"711"/.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:42
S11	155	S8 and S10	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:42
S12	2	"20040230750".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:54
S13	1	S2 and S12	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:54
S14	2	S2 and S6	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 07:56
S15	2	"20050226083".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/06/12 08:00


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

 Terms used **cluster node multiprocessor coherence one to one**

Found 57 of 185,030

Sort results by


[Save results to a Binder](#)
[Try an Advanced Search](#)
[Try this search in The ACM Guide](#)

Display results


[Search Tips](#)
☐ Open results in a new window

Results 1 - 20 of 57

 Result page: [1](#) [2](#) [3](#) [next](#)

 Relevance scale ☐ ☐ ☐ ☐ ☐

1 [Shared memory computing on clusters with symmetric multiprocessors and system area networks](#)



Leonidas Kontothanassis, Robert Stets, Galen Hunt, Umit Rencuzogullari, Gautam Altekar, Sandhya Dwarkadas, Michael L. Scott

 August 2005 **ACM Transactions on Computer Systems (TOCS)**, Volume 23 Issue 3

Publisher: ACM Press

 Full text available: [pdf\(918.28 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Cashmere is a software distributed shared memory (S-DSM) system designed for clusters of server-class machines. It is distinguished from most other S-DSM projects by (1) the effective use of fast user-level messaging, as provided by modern system-area networks, and (2) a "two-level" protocol structure that exploits hardware coherence within multiprocessor nodes. Fast user-level messages change the tradeoffs in coherence protocol design; they allow Cashmere to employ a relatively simp ...

Keywords: Distributed shared memory, relaxed consistency, software coherence

2 [The directory-based cache coherence protocol for the DASH multiprocessor](#)



Daniel Lenoski, James Laudon, Kourosh Gharachorloo, Anoop Gupta, John Hennessy

 May 1990 **ACM SIGARCH Computer Architecture News , Proceedings of the 17th annual international symposium on Computer Architecture ISCA '90**, Volume 18 Issue 3a

Publisher: ACM Press

 Full text available: [pdf\(1.74 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

DASH is a scalable shared-memory multiprocessor currently being developed at Stanford's Computer Systems Laboratory. The architecture consists of powerful processing nodes, each with a portion of the shared-memory, connected to a scalable interconnection network. A key feature of DASH is its distributed directory-based cache coherence protocol. Unlike traditional snoopy coherence protocols, the DASH protocol does not rely on broadcast; instead it uses point-to-point messages sent between th ...

3 [Cashmere-2L: software coherent shared memory on a clustered remote-write network](#)



Robert Stets, Sandhya Dwarkadas, Nikolaos Hardavellas, Galen Hunt, Leonidas Kontothanassis, Srinivasan Parthasarathy, Michael Scott

October 1997 **ACM SIGOPS Operating Systems Review , Proceedings of the sixteenth ACM symposium on Operating systems principles SOSP '97**, Volume 31 Issue 5

Publisher: ACM Press

Full text available:  [pdf\(2.17 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

4 Code optimization - I: Local scheduling techniques for memory coherence in a clustered VLIW processor with a distributed data cache

Enric Gibert, Jesús Sánchez, Antonio González

March 2003 **Proceedings of the international symposium on Code generation and optimization: feedback-directed and runtime optimization CGO '03**

Publisher: IEEE Computer Society

Full text available:  [pdf\(1.19 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Clustering is a common technique to deal with wire delays. Fully-distributed architectures, where the register file, the functional units and the cache memory are partitioned, are particularly effective to deal with these constraints and besides they are very scalable. However, the distribution of the data cache introduces a new problem: memory instructions may reach the cache in an order different to the sequential program order, thus possibly violating its contents. In this paper two local sch ...

5 The effects of communication parameters on end performance of shared virtual memory clusters

Angelos Bilas, Jaswinder Pal Singh

November 1997 **Proceedings of the 1997 ACM/IEEE conference on Supercomputing (CDROM)**

Publisher: ACM Press

Full text available:  [pdf\(201.86 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

Recently there has been a lot of effort in providing cost-effective Shared Memory systems by employing software only solutions on clusters of high-end workstations coupled with high-bandwidth, low-latency commodity networks. Much of the work so far has focused on improving protocols, and there has been some work on restructuring applications to perform better on SVM systems. The result of this progress has been the promise for good performance on a range of applications at least in the 16-32 pro ...

Keywords: bandwidth, clustering, communication parameters, distributed memory, host overhead, interrupt cost, latency, network occupancy, shared memory

6 Communication and consistency protocols: Detailed cache coherence characterization for OpenMP benchmarks

Jaydeep Marathe, Anita Nagarajan, Frank Mueller

June 2004 **Proceedings of the 18th annual international conference on Supercomputing**

Publisher: ACM Press

Full text available:  [pdf\(358.00 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Past work on studying cache coherence in shared-memory symmetric multiprocessors (SMPs) concentrates on studying aggregate events, often from an architecture point of view. However, this approach provides insufficient information about the exact sources of inefficiencies in parallel applications. For SMPs in contemporary clusters, application performance is impacted by the pattern of shared memory usage, and it becomes essential to understand coherence behavior in terms of the application progra ...

Keywords: SMPs, cache analysis, coherence protocols, dynamic binary rewriting, program instrumentation

7 The performance of cache-coherent ring-based multiprocessors



Luis André Barroso, Michel Dubois

May 1993 **ACM SIGARCH Computer Architecture News , Proceedings of the 20th annual international symposium on Computer architecture ISCA '93**, Volume 21 Issue 2

Publisher: ACM Press

Full text available: pdf(1.03 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Advances in circuit and integration technology are continuously boosting the speed of microprocessors. One of the main challenges presented by such developments is the effective use of powerful microprocessors in shared memory multiprocessor configurations. We believe that the interconnection problem is not solved even for small scale shared memory multiprocessors, since the speed of shared buses is unlikely to keep up with the bandwidth requirements of new microprocessors. In this paper we ...

8 Parallel simulation of chip-multiprocessor architectures



Matthew Chidester, Alan George

July 2002 **ACM Transactions on Modeling and Computer Simulation (TOMACS)**, Volume 12 Issue 3

Publisher: ACM Press

Full text available: pdf(519.20 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Chip-multiprocessor (CMP) architectures present a challenge for efficient simulation, combining the requirements of a detailed microprocessor simulator with that of a tightly-coupled parallel system. In this paper, a distributed simulator for target CMPs is presented based on the Message Passing Interface (MPI) designed to run on a host cluster of workstations. Microbenchmark-based evaluation is used to narrow the parallelization design space concerning the performance impact of distributed vs. ...

Keywords: Chip multiprocessors (CMP); Myrinet, Scalable Coherent Interface (SCI), microbenchmarks

9 A survey of commercial parallel processors



Edward Gehringer, Janne Abullarade, Michael H. Guly

September 1988 **ACM SIGARCH Computer Architecture News**, Volume 16 Issue 4

Publisher: ACM Press

Full text available: pdf(2.96 MB)

Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This paper compares eight commercial parallel processors along several dimensions. The processors include four shared-bus multiprocessors (the Encore Multimax, the Sequent Balance system, the Alliant FX series, and the ELXSI System 6400) and four network multiprocessors (the BBN Butterfly, the NCUBE, the Intel iPSC/2, and the FPS T Series). The paper contrasts the computers from the standpoint of interconnection structures, memory configurations, and interprocessor communication. Also, the share ...

10 Dynamic node reconfiguration in a parallel-distributed environment




Michael J. Feeley, Brian N. Bershad, Jeffrey S. Chase, Henry M. Levy

April 1991 **ACM SIGPLAN Notices , Proceedings of the third ACM SIGPLAN symposium on Principles and practice of parallel programming PPOPP '91**, Volume 26 Issue 7

Publisher: ACM Press

Full text available:  pdf(901.48 KB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

11 Memory coherence activity prediction in commercial workloads

 Stephen Somogyi, Thomas F. Wenisch, Nikolaos Hardavellas, Jangwoo Kim, Anastassia Ailamaki, Babak Falsafi

June 2004 **Proceedings of the 3rd workshop on Memory performance issues: in conjunction with the 31st international symposium on computer architecture WMPI '04**


Publisher: ACM Press

Full text available:  pdf(434.65 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Recent research indicates that prediction-based coherence optimizations offer substantial performance improvements for scientific applications in distributed shared memory multiprocessors. Important commercial applications also show sensitivity to coherence latency, which will become more acute in the future as technology scales. Therefore it is important to investigate prediction of memory coherence activity in the context of commercial workloads. This paper studies a trace-based Downgrade Predi ...


Keywords: coherence misses, coherence prediction, commercial workloads, sharing patterns, trace-based prediction

12 Cache inclusion and processor sampling in multiprocessor simulations

 Jacqueline Chame, Michel Dubois


June 1993 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1993 ACM SIGMETRICS conference on Measurement and modeling of computer systems SIGMETRICS '93**, Volume 21 Issue 1

Publisher: ACM Press

Full text available:  pdf(1.17 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)


The evaluation of cache-based systems demands careful simulations of entire benchmarks. Simulation efficiency is essential to realistic evaluations. For systems with large caches and large number of processors, simulation is often too slow to be practical. In particular, the optimized design of a cache for a multiprocessor is very complex with current techniques. This paper addresses these problems. First we introduce necessary and sufficient conditions for cache inclusion in systems with invalid ...

13 Real-time shading

 Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes SIGGRAPH '04**

Publisher: ACM Press

Full text available:  pdf(7.39 MB) Additional Information: [full citation](#), [abstract](#)

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...

14 Parallel execution of prolog programs: a survey

Gopal Gupta, Enrico Pontelli, Khayri A.M. Ali, Mats Carlsson, Manuel V. Hermenegildo

July 2001 **ACM Transactions on Programming Languages and Systems (TOPLAS)**,

Volume 23 Issue 4

Publisher: ACM Press

Full text available: pdf(1.95 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Since the early days of logic programming, researchers in the field realized the potential for exploitation of parallelism present in the execution of logic programs. Their high-level nature, the presence of nondeterminism, and their referential transparency, among other characteristics, make logic programs interesting candidates for obtaining speedups through parallel execution. At the same time, the fact that the typical applications of logic programming frequently involve irregular computatio ...

Keywords: Automatic parallelization, constraint programming, logic programming, parallelism, prolog

15 Formal verification in hardware design: a survey

Christoph Kern, Mark R. Greenstreet

April 1999 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**,

Volume 4 Issue 2

Publisher: ACM Press

Full text available: pdf(411.53 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In recent years, formal methods have emerged as an alternative approach to ensuring the quality and correctness of hardware designs, overcoming some of the limitations of traditional validation techniques such as simulation and testing. There are two main aspects to the application of formal methods in a design process: the formal framework used to specify desired properties of a design and the verification techniques and tools used to reason about the relationship between a spec ...

Keywords: case studies, formal methods, formal verification, hardware verification, language containment, model checking, survey, theorem proving

16 The Mercury Interconnect Architecture: a cost-effective infrastructure for high-performance servers

Wolf-Dietrich Weber, Stephen Gold, Pat Helland, Takeshi Shimizu, Thomas Wicki, Winfried Wilcke

May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture ISCA '97**, Volume 25 Issue 2

Publisher: ACM Press

Full text available: pdf(1.53 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


This paper presents HAL's Mercury Interconnect Architecture, an interconnect infrastructure designed to link commodity microprocessors, memory, and I/O components into high-performance multiprocessing servers. Both shared-memory and message-passing systems, as well as hybrid systems are supported by the interconnect. The key attributes of the Mercury Interconnect Architecture are: low latency, high bandwidth, a modular and flexible design, reliability/availability/serviceability (RAS) features, ...

17 Visual navigation of large environments using textured clusters

Paulo W. C. Maciel, Peter Shirley

April 1995 **Proceedings of the 1995 symposium on Interactive 3D graphics**

Publisher: ACM Press

Full text available:  [pdf\(2.93 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

A visual navigation system is described which uses texture mapped primitives to represent clusters of objects to maintain high and approximately constant frame rates. In cases where there are more unoccluded primitives inside the viewing frustum than can be drawn in real-time on the workstation, this system ensures that each visible object, or a cluster that includes it, is drawn in each frame. The system supports the use of traditional "level-of-detail" representations for indi ...

18 Memory access buffering in multiprocessors



M. Dubois, C. Scheurich, F. Briggs

June 1986 **ACM SIGARCH Computer Architecture News , Proceedings of the 13th annual international symposium on Computer architecture ISCA '86**, Volume 14 Issue 2

Publisher: IEEE Computer Society Press, ACM Press

Full text available:  [pdf\(943.66 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In highly-pipelined machines, instructions and data are prefetched and buffered in both the processor and the cache. This is done to reduce the average memory access latency and to take advantage of memory interleaving. Lock-up free caches are designed to avoid processor blocking on a cache miss. Write buffers are often included in a pipelined machine to avoid processor waiting on writes. In a shared memory multiprocessor, there are more advantages in buffering memory requests, since each m ...


19 Techniques for reducing consistency-related communication in distributed shared-memory systems



John B. Carter, John K. Bennett, Willy Zwaenepoel

August 1995 **ACM Transactions on Computer Systems (TOCS)**, Volume 13 Issue 3

Publisher: ACM Press

Full text available:  [pdf\(2.86 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Distributed shared memory (DSM) is an abstraction of shared memory on a distributed-memory machine. Hardware DSM systems support this abstraction at the architecture level; software DSM systems support the abstraction within the runtime system. One of the key problems in building an efficient software DSM system is to reduce the amount of communication needed to keep the distributed memories consistent. In this article we present four techniques for doing so: software release consistency; m ...


Keywords: cache consistency protocols, distributed shared memory, memory models, release consistency, virtual shared memory

20 Is data distribution necessary in OpenMP?

Dimitrios S. Nikolopoulos, Theodore S. Papatheodorou, Constantine D. Polychronopoulos, Jesus Labarta, Eduard Ayguade;

November 2000 **Proceedings of the 2000 ACM/IEEE conference on Supercomputing (CDROM)**

Publisher: IEEE Computer Society

Full text available:  [pdf\(116.52 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)[Publisher Site](#)

This paper investigates the performance implications of data placement in OpenMP programs running on modern ccNUMA multiprocessors. Data locality and minimization of the rate of remote memory accesses are critical for sustaining high performance on these systems. We show that due to the low remote-to-local memory access latency ratio of

state-of-the-art ccNUMA architectures, reasonably balanced page placement schemes-such as round-robin or random distribution of pages-incur modest performa ...

Results 1 - 20 of 57

Result page: [1](#) [2](#) [3](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)